Introduction to GitHub

Link to Download Git Client https://git-scm.com/downloads

Introduction

- Why learn this?
- Why version control?
  - Save different version of code – note stable checkpoints
  - Work on multiple independent feature simultaneously
  - Multiple people working on same codebase simultaneously
- History, Etymology
- What is Git? Why Git?
- What is GitHub? Why GitHub? [Others: Gitlab, Bitbucket]
- Git Features
  - Non-linear development
  - Distributed Development
  - Compatibility with existing systems and protocols

Flow

- Create New Git Repository(Always give a meaningful name to a repository/folder)
  - $git clone <repository-address>
  - Make a local directory(Not necessarily same name but same is preferred) and then do $git init
- Workflow for committing
  - Working Directory

  - Index(Stage)
  - HEAD
- Add & Commit
  - git add <filename>
  - git add *
  - git add .
  - git commit -m "Commit Message" [Stable checkpoints]
- Pushing Changes
  - git remote add origin <remote-address> (If not cloned)
  - git push origin master (If clone we just need to $git push)
- Branching
  - git checkout -b feature_x
  - git checkout master
  - git branch -d feature_x
  - git push origin <branch>

Hands On

Steps : Setting up local repository and pushing the changes

[On Github] https://github.com/

- Create your GitHub account
- Create a repository with README.md file

[On Local Machine- Git Command Prompt]

- $ git config --global user.name "John Doe"
- $ git config --global user.email "johndoe@example.com"
- $git clone or $git init a repository
- If $git clone ()
  - Add file/folder or makes changes to the file
  - $git add [#Show $git status]
  - $git commit -m "Commit Message" [#$git log]
  - $git push [to update the remote]
- If $git init
  - If empty repository - follow below procedure otherwise first pull change with $git pull <remote-repository-address>
  - Add file/folder or makes changes to the file
  - $git add
  - $git commit -m "Commit Message"
  - git remote add origin <remote-repository-address>
  - $git push origin master[just $git push doesn't work]
- $git diff <filename>
- .gitignore [To ignore files and folders]
- Multiple commits and their differences [on GitHub]

Task 1 : Create a remote repository, create a local git repository, add a file in local repository and push changes from local repository to remote repository.

Task 2 : Create a new folder, add a file with some content inside it and push the changes to remote

Task 3 : Create a new folder and add some files in it, create two new file in local repository, Now with .gitignore ignore a file and the folder and push the changes to remote. Finally only one file should be pushed- folder and a file should be ignored.

Reverting to a commit

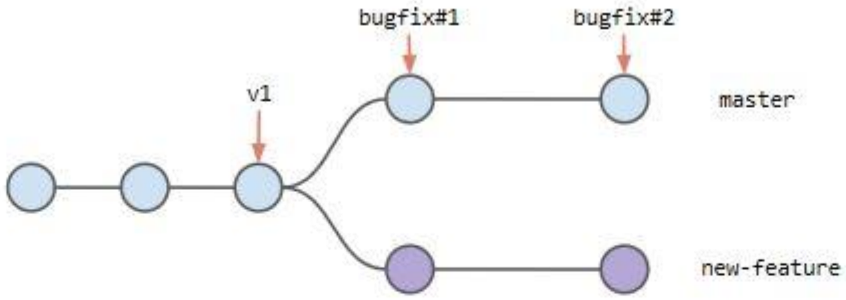
- $git reset –hard
- $git reset –hard <commit-id>

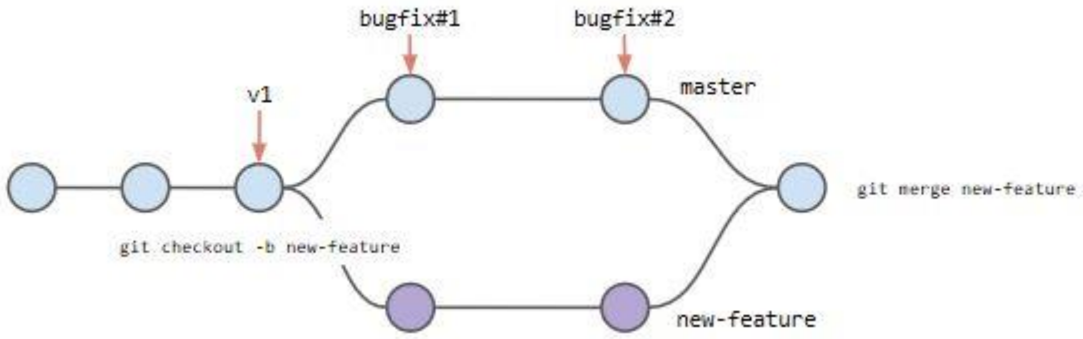
Checkout

$git checkout <commit-id>

$git log --date-order --all --graph --format="%C(green)%h%Creset %C(yellow)%an%Creset %C(blue bold)%ar%Creset %C(red bold)%d%Creset%s"

Branch

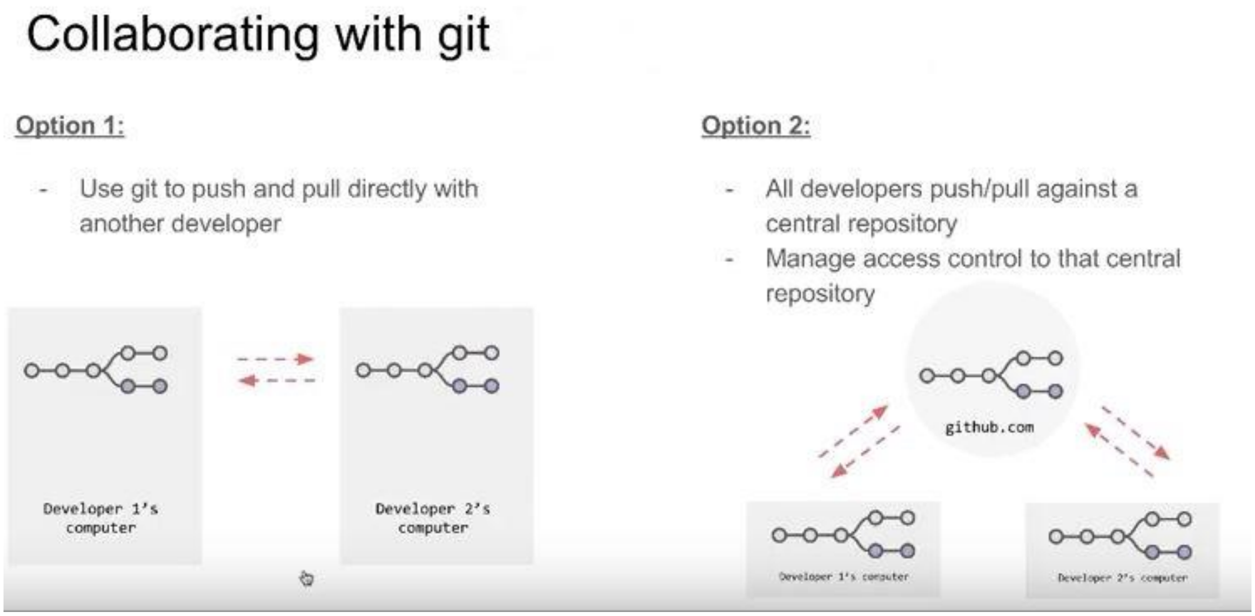




$git checkout -b <new-branch> [Creates a new branch and switches to it]

```
[Make changes in new branch]

$git checkout master

[Make changes in master branch]

$git push origin master

$git push origin <new-branch>

$git pull origin master

$git checkout <new-branch>

$git merge master

$git checkout master

$git merge <new-branch>

$git push origin master
```

<u>Distributed Development</u>



Collaborating with git

Option 1:

- Use git to push and pull directly with another developer

Option 2:

- All developers push/pull against a central repository
- Manage access control to that central repository

Task 4: Make a group of 2 and demonstrate the collaborative development process on GitHub.

Other Points

- Difference between Head and Master
- Difference between git pull and git clone
- .git file
- .gitignore https://git-scm.com/docs/gitignore
- git status
- git log [To view history of commits]
- How to add a folder?
- Downloading the GitHub repository is not equivalent to git clone or git pull

Reference/Resources

https://education.github.com/pack GitHub Student Pack

https://en.wikipedia.org/wiki/Git

http://rogerdudler.github.io/git-guide/

https://git-scm.com/book/en/v2

http://gitimmersion.com/

https://www.atlassian.com/git/tutorials/saving-changes/gitignore

https://help.github.com/articles/getting-started-with-writing-and-formatting-on-github/

https://www.atlassian.com/git/tutorials